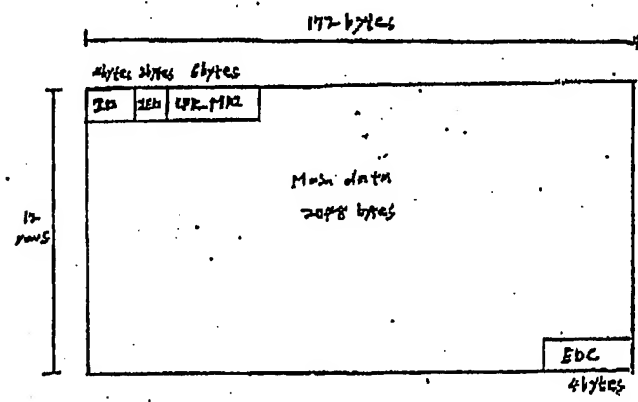




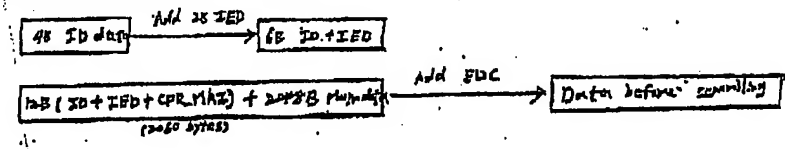
Received: 3/10/03 9:02PM; -->000 ; #725; Page 3 60453194-031103
03/11/03 TUE 09:48 FAX 002.886 2 27552737 TOP TEAM. @003

EDC calculation of DVD Encoding (Error detection code) ①



$$EDC(x) = \sum_{i=0}^6 b_i x^i$$
$$= I(x) \cdot \{9cx\}$$
$$I(x) = \sum_{i=1431}^{15} b_i x^i$$
$$9cx = x^{15} + x^{14} + x^{13} + 1$$

Conventional method



EDC is generated according to ID+IED+CPR_MAI+Main data.
Thus, information of ID,IED,CPR_MAI,Main data should be prepared first before calculating EDC.

Received: 3/10/03 9:02PM; -->000--; #725; Page-4 -- 60453130-031103
 03/11/03 TUE-09:50 FAX-002 886 2 27552737 TOP TEAM 0004

New method

1. calculating a first EDC, which is called as Pre-EDC(PEDC), using ID, IED, CPR_MAI, which are substituted by zero, and original Main
2. calculating a second EDC, which is called as MEDC, using ID, IED, CPR_MAI, and the Main data substituted by zero.
3. calculating real EDC according to the PEDC and MEDC.

$$EDC = PEDC \wedge MEDC$$

Note1: During calculating EDC, 2060 bytes ID+IED+CPR_MAI +Main data are calculated in sequence, consuming DRAM.

Note2: During calculating MEDC, 12 bytes ID+IED+CPR_MAI are calculated, and the 2048 bytes Main data can be passed using a formula or substituted by zero in sequence. Bandwidth of accessing 2048 bytes Main data is saved, but calculation time of MEDC is not saved when substituted by zero (2060T are required when calculating one byte by one byte). DRAM bandwidth and calculation time of MEDC are saved when 2048 bytes of main data is passed by a formula. (13T (12+1) are required when calculating one byte by one byte).

Intention and occasion of using:

1. When main data is downloaded from Host to Host pipe, Host pipe calculates PEDC according to the Main data and stores the PEDC to the DRAM. During encoding, Host pipe calculates MEDC according to ID+IED+CPR_MAI, calculates EDC according to PEDC and MEDC, and stores the EDC to the DRAM. The advantages of this method are that Host pipe receives Main data without the information of the corresponding ID+IED+CPR_MAI, and Encode pipe calculates EDC by reading 12 bytes ID+IED+CPR_MAI and 4 bytes PEDC without reading all the 2060 bytes data, substantially decreasing accessing of DRAM.
2. Some specific data blocks, generally prepared by firmware, might be used repeatedly, thus the main data may be fixed but ID and IED even CPR_MAI may be different. Therefore, new EDC can be generated by calculating PEDC one time and reading corresponding ID, IED, CPR_MAI and PEDC no matter how much times the data block is repeatedly used.

Received: 8/10/03 9:02PM; ->000 ; #728; Page 5
03/11/03 TUE 08:50 FAX 002 886.2 27552737 TOP TEAM

00453194.031103

005

Note 3: ID, IED, CPR_MAI can be substituted by other predetermined values other than zero when calculating PEDC, and Main data can be substituted by other predetermined values other than zero when calculating MEDC.

At this time, $EDC = PEDC \wedge MEDC \wedge C$

Where C is a constant value generated according to the substituted predetermined value.

07

03/10/03 14:33 FAX

3.2 Dec

8012

DVD Specifications for the recordable Disc / Part L Physical Specifications / Version 1.0

3. Information area format

3.2 Data format

3.2.5 Error Detection Code

EDC is a 4-byte check code attached to the 2060 bytes of a Data frame before scrambling.

In [Figure 3.2.1-1], suppose the MSB of the first byte of ID field is b_{31} and the LSB of the last byte of EDC is b_0 , then each bit b_i ($i=31$ to 0) for EDC shall be as follows:

$$\begin{aligned} \text{EDC}(x) &= \sum_{i=0}^n b_i x^i \\ &= I(x) \bmod \{g(x)\} \end{aligned}$$

where,

$$I(x) = \sum_{i=0}^n b_i x^i$$

$$g(x) = x^{32} + x^{31} + x^4 + 1$$